Local Stereo Matching Using Adaptive Local Segmentation

Sanja Damjanović, Ferdinand van der Heijden, Luuk J. Spreeuwers
Signals and Systems Group, Department of EEMCS,
University of Twente, Hallenweg 15, 7522 NH Enschede, The Netherlands

Correspondence should be addressed to Sanja Damjanović s.damjanovic@ewi.utwente.nl

March 24, 2012

Abstract

We propose a new dense local stereo matching framework for gray-level images based on an adaptive local segmentation using a dynamic threshold. We define a new validity domain of the fronto-parallel assumption based on the local intensity variations in the 4-neighborhood of the matching pixel. The preprocessing step smoothes low textured areas and sharpens texture edges, whereas the postprocessing step detects and recovers occluded and unreliable disparities. The algorithm achieves high stereo reconstruction quality in regions with uniform intensities as well as in textured regions. The algorithm is robust against local radiometrical differences; and successfully recovers disparities around the objects edges, disparities of thin objects, and the disparities of the occluded region. Moreover, our algorithm intrinsically prevents errors caused by occlusion to propagate into nonoccluded regions. It has only a small number of parameters. The performance of our algorithm is evaluated on the Middlebury test bed stereo images. It ranks highly on the evaluation list outperforming many local and global stereo algorithms using color images. Among the local algorithms relying on the fronto-parallel assumption, our algorithm is the best ranked algorithm. We also demonstrate that our algorithm is working well on practical examples as for disparity estimation of a tomato seedling and a 3D reconstruction of a face.

1 Introduction

Stereo matching has been a popular topic in computer vision for more than three decades, ever since one of the first papers appeared in 1979 [1]. Stereo images are two images of the same scene taken from different viewpoints. Dense stereo matching is a correspondence problem with the aim to find for each pixel in one image the corresponding pixel in the other image. A map of all pixel displacements in an image is a disparity map. To solve the stereo correspondence problem, it is common to introduce constraints and assumptions, which regularize the stereo correspondence problem.

The most common constraints and assumptions for stereo matching are the epipolar constraint, the constant brightness or the Lambertian assumption, the uniqueness constraint, the smoothness constraint, the visibility constraint and the ordering constraint, [2], [3], [4]. Stereo correspondence algorithms belong to one of two major groups, local or global, depending on whether the constraints are applied to a small local region or propagated throughout the whole image. Local stereo methods estimate the correspondence using a local support region or a window [5], [6]. Local algorithms generally rely on an approximation of the smoothness constraint assuming that all pixels within the matching region have the same disparity. This approximation of the smoothness constraint is known as the fronto-parallel assumption. However, the fronto-parallel assumption is not valid for highly curved surfaces or around disparity discontinuities. Global stereo methods consider stereo matching as a labeling problem where the pixels of the referent image are nodes and the estimated disparities are labels. An energy functional embeds the matching assumptions by its data, smoothness and occlusion terms and propagates them along the scan-line or through the whole image. The labeling problem is solved by energy functional minimization, using dynamic programming, graph cuts or belief propagation [7], [8], [9]. A recent review of both local and global stereo vision algorithms can be found in [10].

Algorithms based on rectangular window matching give an accurate disparity estimation provided the majority of the window pixels belongs to the same, smooth object surface with only a slight curvature or inclination

relative to the image plain. In all other cases, window-based matching produces an incorrect disparity map: the discontinuities are smoothed and the disparities of the high-textured surfaces are propagated into low-textured areas [11]. Another restriction of window-based matching is the size of objects of which the disparity is to be determined; the object's height and width in the image should be at least half the size of the window's dimension in order to be accurately estimated. Algorithms which use suitably shaped matching areas for cost aggregation result in a more accurate disparity estimation, [12], [13], [14], [15], [16], and [17]. The matching region is selected using pixels within certain fixed distances in RGB, CEILab color space and/or Euclidean space.

To alleviate the fronto-parallel assumption, some approaches allow the matching area to lie on the inclined plane, such as in [18] and [19]. The alternative to the idea that properly shaped areas for cost aggregation can result in more accurate matching results is to allocate different weights to pixels in the cost aggregation step. In [20], the pixels closer in the color space and spatially closer to the central pixel are given proportionally more significance, whereas, in [21], the additional assumption of connectivity plays a role during weight assignment.

Our stereo algorithm belongs to the group of local stereo algorithms. Within the stereo framework, we rely on some standard and some modified matching constraints and assumptions. We use the epipolar constraint to convert the stereo correspondence into an one-dimensional problem. However, we modify the interpretation of the fronto-parallel assumption and the Lambertian constraint. A novel interpretation of the fronto parallel assumption is based on local intensity variations. By adaptive local segmentation in both matching windows, we constrain the fronto-parallel assumption only to the intersection of the central matching segments of the initial rectangular window. This mechanism prevents the propagation of the matching errors caused by occlusion and enables an accurate disparity estimation for narrow objects. As only a small subset of window pixels is used for cost calculation, our algorithm is fast and suitable for real-time implementation. The algorithm estimates correctly disparities of both textured as well as textureless surfaces, disparities around depth discontinuities, disparities of the small as well as large objects independently of the initial window size. We apply the Lambertian constraint to local intensity differences and not to the original gray values of the pixels in the segment. In the postprocessing step, we apply the occlusion constraint without imposing the ordering constraint, which enables successful disparity estimation for narrow objects.

Our main contribution is the introduction of the relationship between the fronto-parallel assumption and the local intensity variation and its applications to the stereo matching. In addition, we introduce a preprocessing step that smoothes low textured areas and sharpens texture edges producing the image more favorable for a proper local adaptive segmentation.

The paper is organized as follows: in Section 2, we explain our stereo matching framework: the preprocessing step, the adaptive local segmentation, the matching region selection, the stereo matching, and the postprocessing step; in Section 3, we show and discuss the results of our algorithm on different stereo images; in Section 4 we draw conclusions.

2 Stereo Algorithm

Our algorithm consists of three steps: a preprocessing step, a matching step and a postprocessing step. The flow chart of the algorithm is shown in Figure 1. Input to the algorithm is a pair of rectified stereo images I_l and I_r, where one of them, for instance I_l, is considered as the reference image. For each pixel in the reference image we perform matching along the epipolar line for each integer-valued disparity within the disparity range. Firstly, the input images are preprocessed, as explained in subsection 2.1. The preprocessing step is applied to each image individually. Next, we calculate the local intensity variations maps for the preprocessed images and used them to determine the dynamic threshold for adaptive local segmentation, elaborated in subsection 2.2. Further, the stereo matching comprises a final region selection from segments, a matching cost calculation for all disparities from the disparity range and disparity estimation by a modification of the winer-take-all estimation method, see subsection 2.3. The result of the matching are two disparity maps, D_{LR} and D_{RL}, corresponding to the left and right images of the stereo pair. Finally, postprocessing step calculates the final disparity map corresponding to the reference image as described in subsection 2.4.

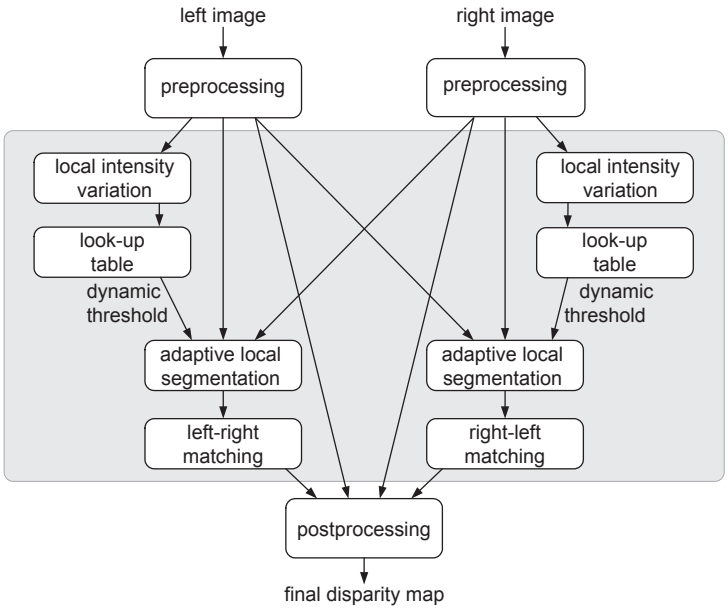


Figure 1: Flow chart of the local stereo matching algorithm using adaptive local segmentation

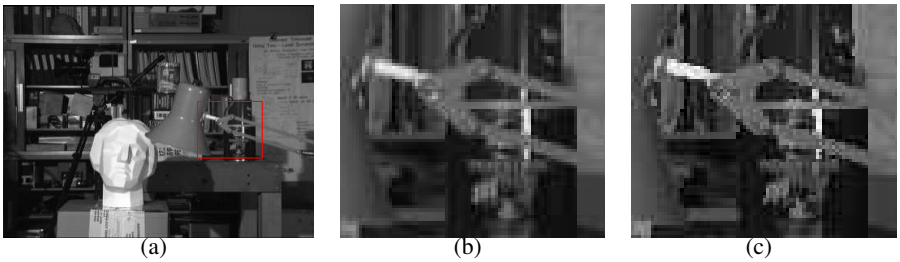


Figure 2: Illustration of the preprocessing step for one image from *Tsukuba* stereo pair: (a) Original image, (b) Detail of the original image, (c) Detail of the original image after the preprocessing step is applied

2.1 Preprocessing

We apply a nonlinear intensity transformation to the input images in order to make them more suitable for adaptive local segmentation. The presence of the Gaussian noise and the sampling errors in image can produce erroneous segments for matching. The noise is dominant in the low textured and uniform regions, while the sampling errors are pronounced in the high textured image regions. The sampling effects can be tackled by choosing a cost measure insensitive to sampling as in [22], or by interpolating the cost function as in [23]. We handle these problems differently and within the preprocessing step. The applied transformation suppresses the noise in low textured regions while simultaneously suppressing the sampling effects in the high textured regions.

The transformation is based on the interpolated subpixel samples by bi-cubic transform in the 4-neighborhood and by consistently replacing the central pixel value by maximum or by minimum value of the set, depending on the relation between the mean and the median of the set. We form a set of samples of the observed pixel at the position (x,y), and the intensities in horizontally and vertically interpolates image at the sub-pixel level at δ_i:

$$\delta_i = -\frac{7}{8} + i \cdot \frac{1}{8}, \;\; i \in \{0, 1, \cdots, 14\}. \tag{1}$$

$$v = \{I(x-\delta_i, y),\, I(x, y-\delta_i) |\, \forall i \in \{0, 1, \cdots, 14\}\}. \quad (2)$$

The sampling effect suppression is performed by replacing the intensity $I(x,y)$ with the new intensity as

$$I(x,y) = \begin{cases} \max\{v\} & : \quad if\ \text{median}\{v\} > \text{mean}\{v\} \\ \min\{v\} & : \quad otherwise \end{cases}$$

All intensity values are corrected in the same manner. If the pixel intensity differs significantly from its four neighbors, as in the high textured regions, it will be replaced by the maximum value in the interpolated subpixel set v, resulting in the sharpening effect. On the other hand, in low textured regions the intensity change is small and replacing the initial intensity value systematically with the minimum value of the interpolated subpixel set v produces favorable denoising effect. These positive effects originate from the image resampling done by bi-cubic interpolation, because the bi-cubic interpolation exhibits overshoots at locations with large differences between adjacent pixels, see chapter 4.4 in [24] and chapter 6.6 in [25]. These favorable effects lack if the interpolation method is linear.

We illustrate the effect of the preprocessing step for an image from a stereo pair from the Middlebury evaluation database in figure 2. Therefore, the preprocessing step modifies regions with high intensity variations and results in the sharper image. Further, in section 3, we show the influence of this step to overall algorithm score.

2.2 Adaptive Local Segmentation

Adaptive local segmentation selects a central subset of pixels from a large rectangular window for which we assume that the fronto parallel assumption holds for the segment. The segment contains the central window pixel and pixels, spatially connected to the central pixel, whose intensities lay within the dynamic threshold from the intensity of the central window. Starting from the segment, we form a final region selection for matching, see subsection 2.3.

The idea behind the adaptive local segmentation is to prevent that the matching region contains the pixels with significantly different disparities prior to actually estimating disparity. We accomplish this aim by conveniently choosing threshold for segmentation based on the local texture. If local texture is uniform with local intensity variations caused only by the Gaussian noise, we opt for a small threshold value. In this way, because the intensity variations are small, the segment will comprise the whole uniform region. We assume that these pixels originate from the smooth surface of one object and therefore that the fronto-parallel assumption holds for the segment. On the other hand, if the window is textured i.e. intensity variations are significantly larger than the noise level, it is not possible to distinguish based only on the pixel intensities and prior to matching, whether the pixels originate from one textured object or from several different objects at different distances from the camera. In this case, relying on the high texture for an accurate matching result, it is good to select small segment in order to assure that the segment contain pixels from only one object and does not contain depth discontinuity. Due to the high local intensity variations, this is achieved by large threshold.

We introduce local intensity variation measure in order to determine the level of local texture and subsequently the dynamic threshold. We define the local intensity variation measure as a sharpness of local edges in the 4-neighborhood of the central window pixel. The sharper local edges are, the local intensity variation is larger. We calculate the local intensity variation using the maximum of the first derivatives in the horizontal and the vertical directions at the half-pixel interpolated imge by benefiting again from overshooting effect of the bi-cubic interpolation.

The horizontal central difference for a pixel at the position (x,y) in image I is calculated as

$$H = \|I(x-\frac{1}{2}, y) - I(x+\frac{1}{2}, y)\|, \quad (3)$$

where $I(x-\frac{1}{2}, y)$ and $I(x+\frac{1}{2}, y)$ are horizontal half-pixel shifts of image I to the left and to the right. The vertical central difference for a pixel at the position (x,y) in image I is calculated as

$$V = \|I(x, y-\frac{1}{2}) - I(x, y+\frac{1}{2})\|, \quad (4)$$

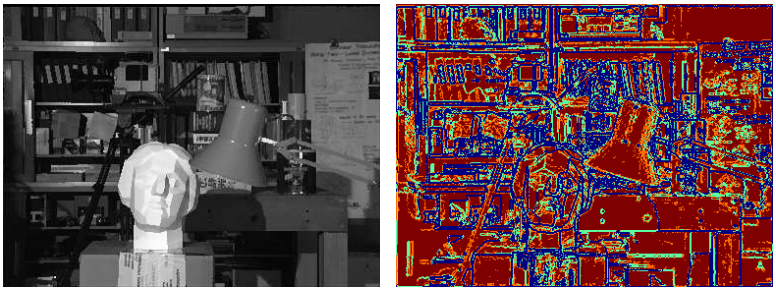


Figure 3: Left image from *Tsukuba* stereo pair with a color-coded local intensity variations levels: the lowest local intensity variation is in red, and in the ascending order follow orange, green, the highest local intensity variations are in blue.

where $I(x,y-\frac{1}{2})$ and $I(x,y+\frac{1}{2})$ are vertical half-pixel shifts of image I. We define the intensity variation measure as

$$M_t(x,y) = max(V,H). \tag{5}$$

We divide local intensity variations into four ranges based on the preselected constant T and define a dynamic threshold for each range by a look-up table :

$$T_d(x,y) = \begin{cases} \frac{T}{2} & : \quad M_t(x,y) \in [0,\frac{T}{4}) \\ \frac{3\cdot T}{4} & : \quad M_t(x,y) \in [\frac{T}{4},\frac{T}{2}) \\ T & : \quad M_t(x,y) \in [\frac{T}{2},T) \\ 2\cdot T & : \quad M_t(x,y) \in [T,\infty) \end{cases} \tag{6}$$

Figure 3 shows a color-coded dynamic threshold map, or equivalently local intensity variation ranges, for the left image from *Tsukuba* stereo pair from the Middlebury stereo evaluation set, [26].

The dynamic threshold $T_d(x,y)$ defined by equation (6) for the referent pixel in the referent image, is also used for the adaptive local segmentation in the non-referent image for all potentially corresponding pixels from the disparity range.

Algorithm 1 Adaptive local segmentation for referent pixel $I_l(x,y)$

Step 1: Dynamic thresholding
for $i = 1$ to W **do**
  for $j = 1$ to W **do**
    if $|wi, j_{l/r} - c_{l/r}| < T_d(x,y)$ **then**
      set $\mathbf{B}_{l/r}^{i,j}$ to 1
    end if
  end for
end for

Step 2: Dilation
Dilate $\mathbf{B}_{l/r}$ with 3×3 squared structured element

Step 3: Imposing connectivity
for $i = 1$ to W **do**
  for $j = 1$ to W **do**
    if $\mathbf{B}_{l/r}^{i,j} = 1$ and not connected to $\mathbf{B}_{l/r}^{w+1,w+1}$ **then**
      set $\mathbf{B}_{l/r}^{i,j}$ to 0
    end if
  end for
end for

The adaptive local segmentation pseudocode for the referent pixel $I_l(x,y)$ in the left image is given by algorithm 1. The segmentation is performed for reference and non-reference windows independently using the same threshold $T_d(x,y)$. Thus, in the $W \times W$ window, where $W = 2 \cdot w + 1$, around the pixel at the position (x,y) in the reference image, we declare that the pixel at (i,j) position, where $i,j = 1,..,W$ in the reference window, belongs to the segment if its gray value $w_l^{i,j}$ differs from the central pixel's gray value $c_l = w_l^{w+1,w+1}$ for less than the dynamic threshold $T_d(x,y)$. The segment pixels in the non-reference window are chosen in similar way using the same threshold $T_d(x,y)$. Next, the central connected components in the dilated masks are selected. The final segments are defined by the binary $W \times W$ maps, B_l and B_r, with ones if the the pixels belong to the segment.

2.3 Stereo Correspondence

The matching region is defined by the overlap of the adaptive local segments in the referent and non-referent windows. Thus, the matching region is defined by binary map B, which has ones if and only if both binary maps, B_l and B_r, have ones at the same positions, as given in algorithm 2.

Algorithm 2 The final binary map calculation

for $i = 1$ to W **do**
  for $j = 1$ to W **do**
    if $\mathbf{B}_l^{i,j} \wedge \mathbf{B}_r^{i,j}$ **then**
      set $\mathbf{B}^{i,j}$ to 1
    end if
  end for
end for

We assume the corresponding pixels have similar intensities and that the differences exist only due to the Gaussian noise with the variance σ_n^2. One-dimensional vectors, $\mathbf{z}_l$ and $\mathbf{z}_r$, are formed from the pixels from the left and right matching window at positions of ones within the binary map $\mathbf{B}$. Besides the noise, differences between vectors can occur due to different offsets and due to occlusion. To make the matching vectors insensitive to local different offsets, we subtract the central pixel values c_l and c_r from vectors $\mathbf{z_l}$ and $\mathbf{z_r}$, given by algorithm 3. In this way, the intensity information is transformed from the absolute intensities to the differences of intensities with respect to the central window pixels. Further, we impose the Lambertian assumtion on the pixels after the central pixel subtraction and not on the original pixel intensities. To prevent the occlusion influence in matching, we eliminate the occlusion outliers by keeping only the coordinates of vectors which differ for less than threshold T as given by algorithm 4.

Algorithm 3 Offset neutralization

N_p' is the length of the vectors $\mathbf{z_l}$ and $\mathbf{z_r}$
c_l and c_r are the central intensities in the left and in the right window
for $i = 1$ to N_p' **do**
  $\mathbf{z}_l(i) = \mathbf{z}_l(i) - c_l$
  $\mathbf{z}_r(i) = \mathbf{z}_r(i) - c_r$
end for

We calculate the matching cost using the sum of squared differences (SSD) [7], [27]. To compare the costs with different length N_p of vectors $\mathbf{z}_l$ and $\mathbf{z}_r$ for different disparities, we introduce the normalized SSD:

$$C_{nSSD} \propto \frac{1}{N_p} \cdot \frac{\| \mathbf{z}_l - \mathbf{z}_r \|^2}{4 \cdot \sigma_n^2}. \tag{7}$$

The *winner-take-all* (WTA) method selects the disparity with the minimal cost for the observed reference pixel. In our algorithm, besides the cost, the number of pixels participating in the cost calculation is also an indication of a correspondence. This ordinal measure cannot be used directly in the disparity estimation,

Algorithm 4 Elimination of the outliers

N'_p is the length of the initial vectors $\mathbf{z}_l$ and $\mathbf{z}_r$
$k = 0$
for $i = 1$ to N'_p **do**
  if $|\mathbf{z}_l(i) - \mathbf{z}_r(i)| < T$ **then**
    Remove $\mathbf{z}_l(i)$ and $\mathbf{z}_r(i)$
  end if
end for
N_p is the length of the final vectors $\mathbf{z}_l$ and $\mathbf{z}_r$

because it is not always a reliable indication of the correspondence as in the case of occlusion. If the number of pixels used in the cost calculation is very low, it may be due to occlusion. However, a reliable match has a substantial ordinal support.

We combine the cost and the number of participating pixels in the disparity estimation and introduce a hybrid WTA: we consider only disparities supported by a sufficient number of pixels as potential candidates for a disparity estimate. Thus, the final disparity estimate is chosen from a subset of the all possible disparities from the disparity range. We term these disparity candidates as the reliable disparity candidates [12], [28].

The reliable disparity candidates have at least $N_s = K_p \cdot \max\{N_p^{x,y}\}$ supporting pixels, where $N_p^{x,y}$ is a set containing the number of pixels participating in the cost aggregation step for each possible disparity value from the disparity range $[D_{min}, D_{max}]$. K_p is the ratio coefficient $0 < K_p \leq 1$. The estimated disparity $d(x,y)$ is:

$$d(x,y) = \underset{d_i \in \{D_{min},\ldots,D_{max}\}}{\arg\min} \{C_{nSSD}^{x,y}(d_i) |\, N_p^{x,y}(d_i) > N_s\}, \tag{8}$$

where $x = 1,\ldots,R$ and $y = 1,\ldots,C$, for image of the dimension $R \times C$ pixels and d_i belongs to the set of all possible disparities from the disparity range $[D_{min}, D_{max}]$.

The final result of the hybrid WTA is the disparity map D

$$D = \{d(x,y) | \forall x \in [1,R] \wedge \forall y \in [1,C]\}. \tag{9}$$

We calculate two disparity maps, one disparity map, D_{LR}, with the left image I_l as the referent, and the other, D_{RL}, as the right image I_r as the reference.

2.4 Postprocessing

In the postprocessing, we detect the disparity errors and correct them. There are some areas of incorrect disparity values caused by low textured areas larger than the initial window. There are some isolated disparity errors with significantly different disparity from the neighborhood disparities, so called outliers, caused by isolated pixels or groups of several pixels if the adaptive local segmentation did not result in sufficiently large segment due to high local intensity variation. Also, there are disparity errors caused by occlusion. Although the matching procedure is the same for both occluded and nonoccluded pixels, our stereo matching algorithm does not propagate error caused by occlusions because the boundaries of objects are taken into account by both the adaptive local segmentation and the final matching region selection. However, occluded pixels do not have corresponding pixels and the estimated disparities for the occluded pixels are incorrect. The post-processing consists of several steps including median filtering of the initial disparity maps, disparity refinement of the individual disparity maps, consistency check and propagation of the reliable disparities.

First, we apply $L \times L$ *median filter* to both disparity maps, D_{LR} and D_{RL}, and eliminate disparity outliers. Second, we refine the filtered disparity maps individually to correct low textures areas with erroneous disparities, in an iterative procedure. The refinement step propagates disparities by histogram voting to the regions with close intensities defined by a look-up table given in equation (10) across the whole image as illustrated in propagation scheme in figure 4. Some similar notions to this approach appear separately in the literature, [17] and [29], and we were inspired by them. In [29], the cost aggregation is done along the 16 radial directions in disparity space, while in [17], histogram voting is used within the segment for disparity refinement. We

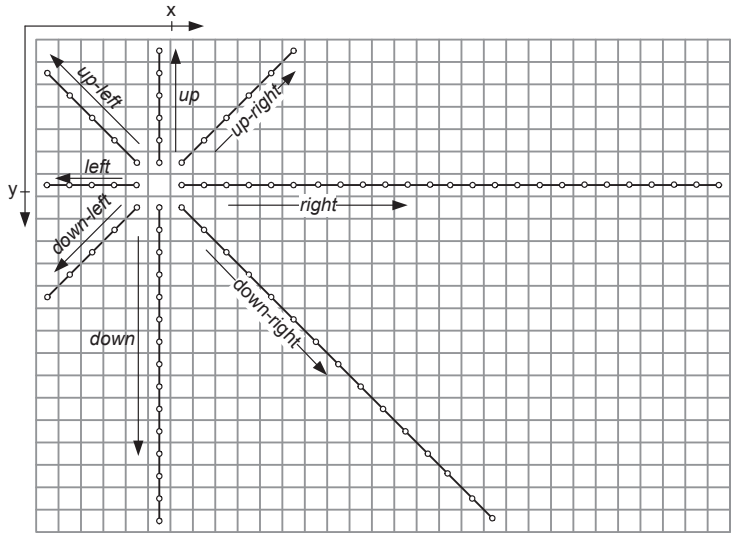


Figure 4: Propagation scheme

Table 1: x_{tmp} and y_{tmp} values for histogram calculation in equation (11)

	direction	x_{tmp}	y_{tmp}	condition
1	up	x	$y-i_u$	$i_u = \{\ 1 \ \text{ to } \ y-1 \mid y-1>0\ \}$
2	up-right	$x-i_{ur}$	$y+i_{ur}$	$i_{ur} = \{\ 1 \ \text{ to } \ \min(x-1, C-y-1) \mid \min(x-1, C-y-1)>0\ \}$
3	right	x	$y+i_r$	$i_r = \{\ 1 \ \text{ to } \ C-y \mid y-C<0\ \}$
4	down-right	$x+i_{dr}$	$y+i_{dr}$	$i_{dr} = \{\ 1 \ \text{ to } \ \min(R-x, C-y) \mid \min(R-x, C-y)>0\ \}$
5	down	$x+i_d$	y	$i_d = \{\ 1 \ \text{ to } \ R-x \mid x<R\ \}$
6	down-left	$x+i_{dl}$	$y-i_{dl}$	$i_{dl} = \{\ 1 \ \text{ to } \ \min(R-x, y-1) \mid \min(R-x, y-1)>0\ \}$
7	left	$x-i_l$	y	$i_l = \{\ 1 \ \text{ to } \ x-1 \mid x-1>0\ \}$
8	up-left	$x-i_{ul}$	$y-i_{ul}$	$i_{ul} = \{\ 1 \ \text{ to } \ \min(x-1, y-1) \mid \min(x-1, y-1)>0\ \}$

refine our disparity maps by histogram voting of accumulating disparities along 8 radial directions across the whole disparity map with constraint of the maximum allowed intensity difference with the pixel being refined. The maximum intensity difference is defined by a dynamic threshold T_p with the same logic behind as in local intensity variation measure in section 2.2, with the difference that here we distinguish three ranges of intensity differences. Thus, the histogram is formed using disparities of the pixels with close intensities along 8 radial directions, see figure 4 and table 1. The intensities are close in intensities and taken into account in histogram forming, if they lie within the threshold T_p from the intensity of the pixel at the observed position (x,y). The threshold $T_p(x,y)$ is selected based on a look-up table:

$$T_p(x,y) = \begin{cases} \frac{T}{2} & : \quad M_t(x,y) \in [0, \frac{T}{2}) \\ \frac{3T}{4} & : \quad M_t(x,y) \in [\frac{T}{2}, \frac{3 \cdot T}{4}) \\ T & : \quad M_t(x,y) \in [\frac{3 \cdot T}{4}, \infty) \end{cases} \tag{10}$$

The histogram H with a number of bins equal to the number of disparities within the disparity range, is formed by counting the disparities along 8 radial directions for the pixels whose intensity is within threshold $T_p(x,y)$:

$$H(d(x_{tmp}, y_{tmp})) = H(d(x_{tmp}, y_{tmp})) + 1, \text{ if } |I(x_{tmp}, y_{tmp}) - I(x,y)| < T_p(x,y), \tag{11}$$

where x_{tmp} and y_{tmp} are given by table 1.

We calculate disparity d_h as a disparity of the normalized histogram maximum:

$$h(i) = \frac{H(i)}{\sum_i H(i)}, \ i = D_{min} \, to \, D_{max} \tag{12}$$

$$d_h = \arg\max_i h(i),\ i = D_{min}\ to\ D_{max} \tag{13}$$

The initial disparity $d(x,y)$ is replaced by the new value d_h if it is significantly supported i.e. if the normalized histogram value $h(d_h)$ is greater than α, otherwise it is left unchanged:

$$d(x,y) = \begin{cases} d_h & : \text{ if } |d_h - d(x,y)| > 1 \ \wedge\ h(d_h) > \alpha \\ d(x,y) & : \text{ otherwise} \end{cases} \tag{14}$$

The steps given by equations (11), (12), (13) and (14), are repeated iteratively until there are no more updates to disparities in the map.

Next, we detect *occluded disparities* by the consistency check between two disparity maps:

$$|D_{RL}(x, y - D_{LR}(x,y)) - D_{LR}(x,y)| \leq 1. \tag{15}$$

If the condition in (15) is not satisfied for disparity $D_{LR}(x,y)$, we declare it as inconsistent and eliminate it from the disparity map. The missing disparities are filled in by an iterative refinement procedure similar to the previously applied procedure for the disparity propagation by histogram voting. In the iterative step to fill in the inconsistent disparities, we use the threshold look-up table (10) as in the disparity refinement step. We calculate the histogram h of the consistent disparities with close intensities along 8 radial directions as given by (11) and (12). The missing disparity is filled in with the disparity d_h with the largest support in the histogram, provided that the histogram is not empty. The remaining unfilled inconsistent disparities, we fill in by the disparity of the nearest neighbor with known disparities with the smallest intensity differences. As a last step in the postprocessing, we apply $L \times L$ *median filter* to obtain the final disparity map.

3 Experiments and Discussion

We have used the Middlebury stereo benchmark [4] to evaluate the performance of our stereo matching algorithm. The parameters of the algorithm are fixed for all four stereo pairs as required by the benchmark. The threshold value is set to $T = 12$. The half-window size is $w = 15$, and the window size is $W \times W$ where $W = 31$. The noise variance σ_n^2 is a small and constant scaling factor in equation (7). The ratio coefficient in hybrid WTA is $K_p = 0.5$. In the post-processing step, the median filter parameter is $L = 5$ and the histogram voting parameter is $\alpha = 0.45$.

Figure 5 shows results for all four stereo pairs from the Middlebury stereo evaluation database: *Tsukuba*, *Venus*, *Teddy* and *Cones*. The leftmost column contains the left images of the four stereo pairs. The ground truth (GT) disparity maps are shown in the second column, the estimated disparity maps are shown in the third column and the error maps are shown in the forth column. In the error maps, the white regions denote correctly calculated disparity values which do not differ for more than 1 from the ground truth. If the estimated disparity differs for more than 1 from the ground truth value, it is marked as an error. The errors are shown in black and gray, where black represents the errors in the nonoccluded regions and gray represents errors in the occluded regions. The quantitative results in the Middlebury stereo evaluation framework are presented in Table 2.

The results show that our stereo algorithm preserves disparity edges. It estimates successfully the disparities of thin objects, and successfully deals with subtle radiometrical differences between images of the same stereo pair. Occlusion errors are not propagated and occluded disparities are successfully filled in the post-processing step. A narrow object is best visible in the *Tsukuba* disparity map (the lamp construction) and in *Cones* disparity map (pens in a cup in the lower right corner). Our algorithm correctly estimates disparities of both textureless and textured surfaces e.g. the example of large uniform surfaces in stereo pairs *Venus* and *Teddy* are successfully recovered.

The images in the Middlebury database have different sizes, different disparity ranges, and different radiometric properties. The stereo pairs *Tsukuba*, 384×288 pixels, and *Venus*, 434×383 pixels, have disparity ranges from 0 to 15 and from 0 to 19. The radiometric properties of the images in these stereo pairs are almost identical, and the offset compensation given by algorithm 3 is not significant for these two example pairs, as we demonstrated in [12]. As required by the Middlebury evaluation framework, we apply the offset compensation to all four stereo pairs. The stereo pairs *Teddy*, 450×375 pixels, and *Cones*, 450×375 pixels, have disparity ranges from 0 to 59. The images of these stereo pairs are not radiometrically identical and the offset compensation successfully deals with these radiometrical differences [12].

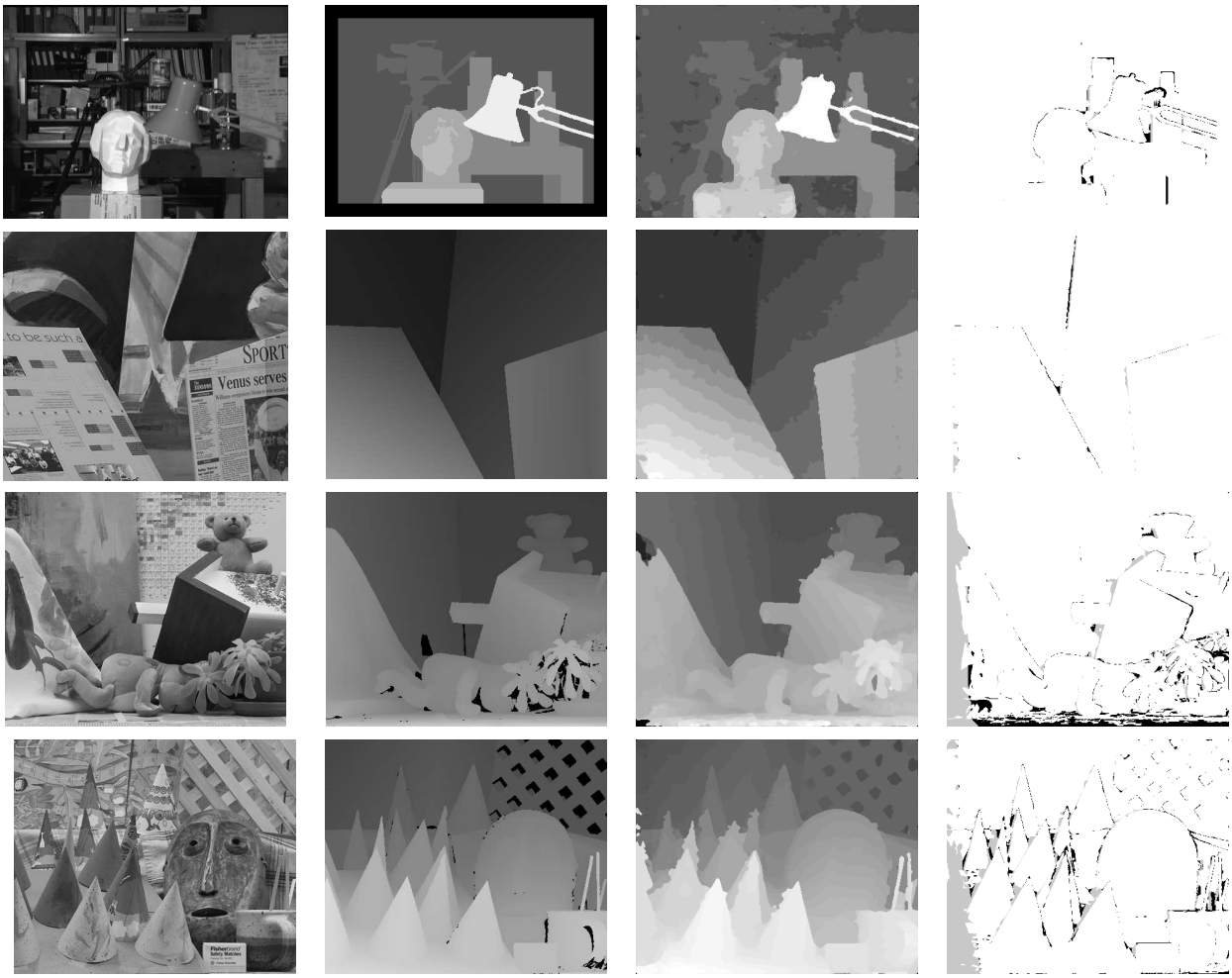


Figure 5: Disparity results for the stereo pairs (1st row: *Tsukuba*, 2nd row: *Venus*, 3rd row: *Teddy*, 4th row: *Cones*) from the Middlebury testbed database. The columns show, from left to the right : The left image, Ground truth, Result computed by our stereo algorithm, Disparity error map larger than 1 pixel. The nonoccluded regions errors with ranking, on March 23^{rd}, 2012, are respectively: *Tsukuba* 1.33% (37), *Venus* 0.32% (39), *Teddy* 5.32% (17), *Cones* 2.73% (14)

Table 2: Evaluation results based on the online Middlebury stereo benchmark [4]: The errors are given in percentages for the nonoccluded (NONOCC) region, the whole image (ALL) and discontinuity (DISC) areas. The numbers within brackets indicate the ranking in the Middlebury table on March 23^{rd}, 2012.

Images	NONOCC	ALL	DISC
Tsukuba	**1.33 (37)**	1.82 (32)	7.19 (46)
Venus	**0.32 (39)**	0.79 (46)	4.5 (58)
Teddy	**5.32 (17)**	11.9 (40)	14.5 (19)
Cones	**2.73 (14)**	9.69 (53)	7.91 (21)

The error percentages together with ranking in the Middlebery evaluation online list are given in Table 2. The numbers show error percentages for non-occluded regions (NONOCC), discontinuity regions (DISC) and the whole (ALL) disparity map. The overall ranking of our algorithm in the Middlebery evaluation table of stereo algorithms is the 28^{th} place out of 123 evaluated algorithms. Thus, our stereo algorithm outperforms many local as well as global algorithms. Among the algorithms ranked in the Middlebury stereo evaluation, there are only two local algorithms ranked higher than our algorithm but both of them do not impose the fronto-parallel assumption strictly: a local matching method using image geodesic supported weights *GeoSup* from [5], and a matching approach with slanted support windows PatchMatch in [30]. Both of these algorithms use colored images, while our algorithm works with intensity images and achieves comparable results. Although these approaches have better general ranking in the Middlebury stereo evaluation list, our approach with matching based on fronto-parallel regions outperforms the *PatchMatch* algorithm for *Tsukuba* stereo pair, and the *GeoSup* algorithm for *Tsukuba*, *Teddy* and *Cones* stereo pairs. Thus, our approach with region selection by threshold produces more accurate disparity maps for cluttered scenes than *GeoSup* algorithm with region selection using geodesic support weights.

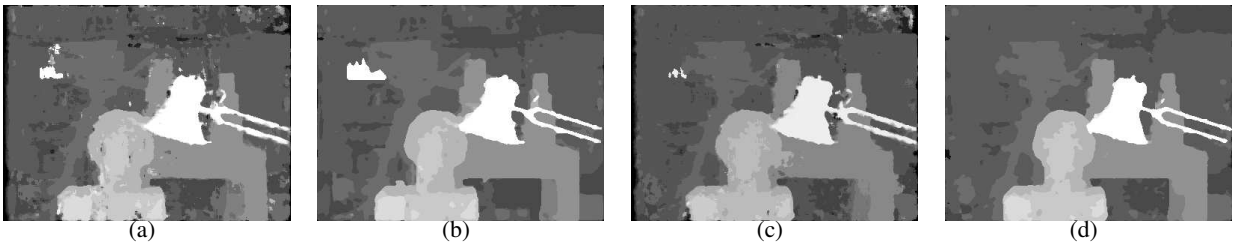
To investigate the contribution of the preprocessing and the postprocessing steps to the overall result, we show in table 3 the results we obtained on the benchmark stereo pairs with or without the preprocessing and the postprocessing steps in the algorithm. We show the results if neither, only one, and both steps are applied. If our postprocessing step was omitted, the $L \times L$ median filter was applied. From the results in table 3, we conclude that both steps, if individually applied, improve the qualities of the final disparity maps. If we apply both steps, the accuracy of the disparity maps is the highest. Furthermore, the improvement contribution of the preprocessing step is greater than the postprocessing step only for *Venus* stereo pair. This is because the sampling effects were most pronounced in *Venus* scene. In addition, we show in figure 6 the disparity maps for *Tsukuba* stereo pair for all four combinations: if the preprocessing and the postprocessing steps are included or not in the algorithm. We conclude that the preprocessing step plays a significant role in accurate disparity estimation of textureless areas, while the postprocessing step especially helps in an accurate estimation of disparity discontinuities.

To illustrate the subtle features of our algorithm not captured in the standard test bed images, we apply our stereo algorithm, while retaining the parameter values, on some other images from the Middlebury site in Figure 7. For two other stereo pairs, *Art* and *Dolls*, we show the left images of two stereo pairs in the leftmost column. The ground truth (GT) disparity maps are in the second column. The third column shows our estimation of the disparity maps. The fourth column shows the error maps with regard to the ground truth. The algorithm successfully recovers the disparities of very narrow structures as in *Art* disparity map. The disparity of the cluttered scene is successfully estimated, as in *Dolls* disparity map.

		Tsukuba			*Venus*			*Teddy*			*Cones*		
preP	postP	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc
-	-	3.6	5.41	10.04	2.76	4.38	13.18	8.11	17.42	19.73	4.77	15.04	12.33
+	-	2.74	4.50	10.11	0.62	1.63	7.95	7.52	16.82	19.41	3.98	14.37	11.27
-	+	2.45	3.05	7.31	1.53	2.11	5.75	6.11	12.49	15.20	3.20	9.30	9.14
+	+	1.33	1.82	7.19	0.32	0.79	4.5	5.32	11.90	14.50	2.73	9.69	7.91

Table 3: Comparison of results with (+) or without(-) preprocessing (preP) and postprocessing (postP) steps

Next, we demonstrate that the presented local stereo algorithm works well on practical problems. Examples



(a) (b) (c) (d)

Figure 6: Disparity results for the stereo pair *Tsukuba*: (a) without preprocessing and without postprocessing, (b) without preprocessing and with postprocessing, (c) with preprocessing and without postprocessing, (d) with preprocessing and with postprocessing

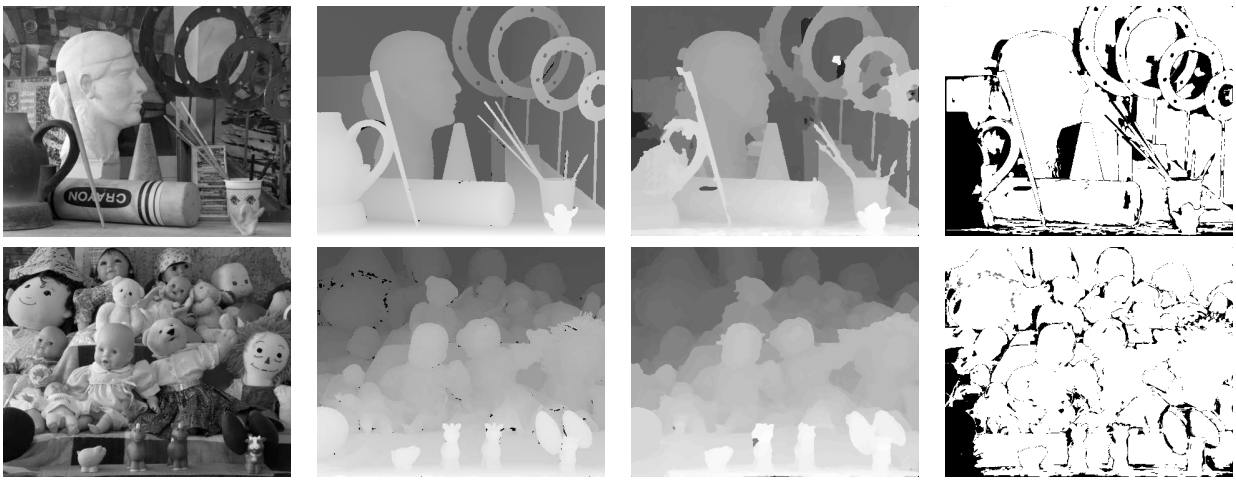
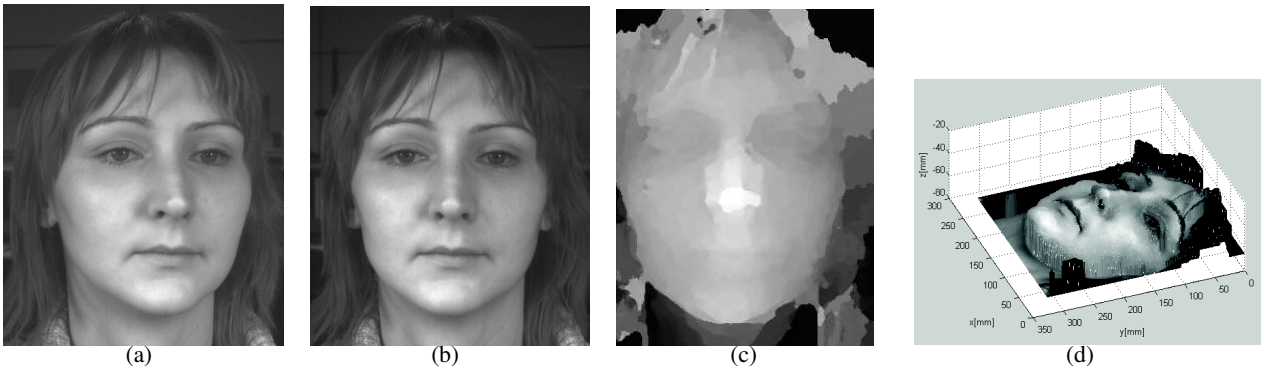
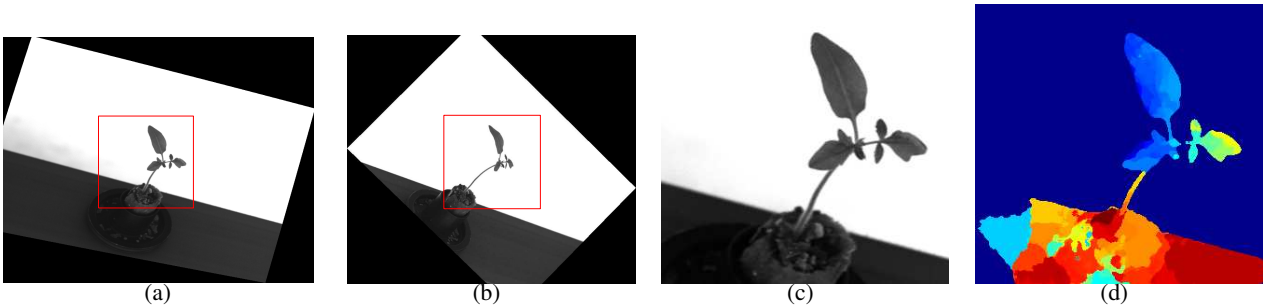


Figure 7: Disparity results for the stereo pairs (1st row: *Art*, 2nd row: *Dolls*) from the Middlebury database of the stereo images. Size of each image is 370×463 pixels. Disparity range in both stereo pair is 0 to 75. The columns show, from left to the right: The left image, The ground truth, The result computed by our stereo algorithm, The disparity error map larger than 1 pixel.



(a) (b) (c) (d)

Figure 8: Disparity results for the stereo pair *Sanja*, taken at the vision laboratory of Signals and Systems Group, University of Twente. Size of each image is 781×641 pixels. Disparity range is 0 to 40. (a) Left stereo image (b) Right stereo image (c) Disparity map corresponding to the right image (d) Depth map with texture overlay



(a) (b) (c) (d)

Figure 9: Disparity results for the stereo pair *Tomato seedling*, taken within MARVIN project at the vision laboratory of Intelligent System Group, Wageningen UR - Food and Biobased Research. Size of the region of interest in each image is 300×310 pixels. Disparity range is 0 to 90. (a) Left stereo image (b) Right stereo image (c) Region of interest in the left image (d) Disparity map corresponding to the left image

of disparity map estimation and 3D reconstruction of a face are shown for stereo pair *Sanja* in figure 8. The disparity map estimation of a plant in stereo pair *Tomato seedling* is shown in figure 9. The parameters of the algorithm are kept the same as in the previous examples. Thus, our algorithm successfully estimates the disparity of the smooth low textured objects and is suitable also for application to 3D face reconstruction, figure 8(d). Our algorithm also successfully estimated the disparity map of the tomato seedling. *Tomato seedling* stereo images represent a challenging task for a stereo matching algorithm in general, because the viewpoints significantly differ and the structure of the plant is narrow i.e. much smaller than the window dimension.

As far as the initial window size is concerned, our algorithm is not influenced by the window size above certain size. In principle, we could apply our algorithm using the whole image as the initial window around the reference pixel. This would result in a sufficiently large region selection for uniform regions in the image and make the the ordinal measure within the hybrid WTA more reliable. On the other hand, in matching windows with high local intensity variations, the selected region is always significantly smaller than the window and does not change if the window is enlarged because of the connectivity constraint with the referent central pixel.

4 Conclusion

In our local stereo algorithm, we have introduced a new approach for stereo correspondence based on the adaptive local segmentation by a dynamic threshold so that the fronto-parallel assumption holds for a segment. Further, we have established a relationship among the local intensity variation in an image and the dynamic

threshold. We have applied the novel preprocessing procedure on both stereo images to eliminate the influence of noise and sampling artifacts. The mechanism for the final matching region selection prevents error propagation due to disparity discontinuities and occlusion. In the postprocessing step, we introduce a new histogram voting procedure for disparity refinement and for filling in the eliminated inconsistent disparities. Although, the starting point in matching is the large rectangular window, disparity of narrow structures is accurately estimated.

We evaluated our algorithm on the stereo pairs from the Middlebury database. It ranks highly on the list, outperforming many local and global algorithms that use color information while we use only intensity images. Our algorithm is the best performing algorithm in the class of local algorithms which use intensity images and the fronto-parallel assumption without weighting the intensities of the matching region. Furthermore, our algorithm matches textureless as well as textured surfaces equally well, handles well the local radiometric differences, preserves edges in disparity maps, and successfully recovers the disparity of thin objects and the disparities of the occluded regions. We demonstrated the performance of our algorithm on two additional examples from the Middlebury database and on two practical examples. The results on this additional examples show that the disparity maps of scenes of different natures are successfully estimated: smooth low textured objects as well as textured cluttered scenes, narrow structures and textureless surfaces. Moreover, our algorithm has also other positive aspects making it suitable for real time implementation: it is local; it has just three parameters; intensity variations are locally calculated and there is no global segmentation algorithm involved.

References

[1] D. Marr and T.A. Poggio. A computational theory of human stereo vision. *Proceedings of the Royal Society of London*, B-204:301–328, 1979.

[2] M. Z. Brown, D. Burschka, and G. D. Hager. Advances in computational stereo. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(8):993–1008, 2003.

[3] O.D. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, 1993.

[4] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1-3):7–42, 2002.

[5] A. Hosni, M. Bleyer, M. Gelautz, and C. Rhemann. Local stereo matching using geodesic support weights. In *IEEE Int. Conf. on Image Processing 2009*, 2009.

[6] K. Zhang, J. Lu, and G. Lafruit. Cross-based local stereo matching using orthogonal integral images. *IEEE Trans. Cir. and Sys. for Video Technol.*, 19:1073–1079, July 2009.

[7] P. N. Belhumeur. A Bayesian approach to binocular stereopsis. *Int. J. Comput. Vision*, 19(3):237–260, 1996.

[8] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001.

[9] J. Sun, N.-N. Zheng, and H.-Y. Shum. Stereo matching using belief propagation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(7):787–800, 2003.

[10] L. Nalpantidis, G. C. Sirakoulis, and A. Gasteratos. Review of Stereo Vision Algorithms: from Software to Hardware. *International Journal of Optomechatronics*, 2(4):435–462, 2008.

[11] C. L. Zitnick and T. Kanade. A cooperative algorithm for stereo matching and occlusion detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(7):675–684, 2000.

[12] S. Damjanović, F. van der Heijden, and L. J. Spreeuwers. Sparse window local stereo matching. In *VISIGRAPP 2011*, pages 689–693, 2011.

[13] R. K. Gupta and S.-Y. Cho. Real-time stereo matching using adaptive binarywindow. In *3DPVT 2010*, 2010.

[14] F. Tombari, S. Mattoccia, L. Di Stefano, and E. Addimanda. Classification and evaluation of cost aggregation methods for stereo correspondence. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–8, June 2008.

[15] X. Sun, X. Mei, S. Jiao, M. Zhou, and H. Wang. Stereo matching with reliable disparity propagation. In *IEEE Int. Conf. on 3D Digital Imaging, Modeling, Processing, Visualisation and Transmittion (3DIMPVT)*, 2011.

[16] K. Zhang, J. Lu, and G. Lafruit. Scalable stereo matching with locally adaptive polygon approximation. In *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, pages 313–316, Oct. 2008.

[17] K. Zhang, J. Lu, G. Lafruit, R. Lauwereins, and Luc Van Gool. Accurate and efficient stereo matching with robust piecewise voting. In *Proceedings of the 2009 IEEE international conference on Multimedia and Expo*, ICME'09, pages 93–96, Piscataway, NJ, USA, 2009. IEEE Press.

[18] C. Rother M. Bleyer and P. Kohli. Surface stereo with soft segmentation. In *CVPR*, 2010.

[19] H. Tao, H. S. Sawhney, and R. Kumar. A global matching framework for stereo computation. In *ICCV*, pages 532–539, 2001.

[20] K.-Y. Yoon and I. S. Kweon. Adaptive support-weight approach for correspondence search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):650–656, April 2006.

[21] A. Hosni, M. Bleyer, M. Gelautz, and C. Rhemann. Geodesic adaptive support weight approach for local stereo matching. In *Computer Vision Winter Workshop 2010*, pages 60–65, 2010.

[22] S. Birchfield and C. Tomasi. Depth discontinuities by pixel-to-pixel stereo. *Int. J. Comput. Vision*, 35(3):269–293, 1999.

[23] R. Szeliski and D. Scharstein. Sampling the disparity space image. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(3):419–425, March 2004.

[24] Q. Wu, F. A. Merchant, and K. R. Castleman. *Microscope Image Processing*. Academic Press, 2008.

[25] R. C. Gonzalez, R. E. Woods, and S. L. Eddins. *Digital Image Processing Using MATLAB, 2nd ed.* Gatesmark Publishing, 2nd edition, 2009.

[26] Middlebury stereo website, http://vision.middlebury.edu/stereo/, 2009.

[27] I. J. Cox. A maximum likelihood n-camera stereo algorithm. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 733–739, 1994.

[28] S. Damjanović, F. van der Heijden, and L. J. Spreeuwers. Sparse window local stereo matching. In *Proceedings of the International Workshop on Computer Vision Applications (CVA)*, pages 83–86, 2011.

[29] H. Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(2):328–341, 2008.

[30] B. Bleyer, C. Rhemann, and C. Rother. Patchmatch stereo - stereo matching with slanted support windows. In *British Machine Vision Conference 2011*, 2011.